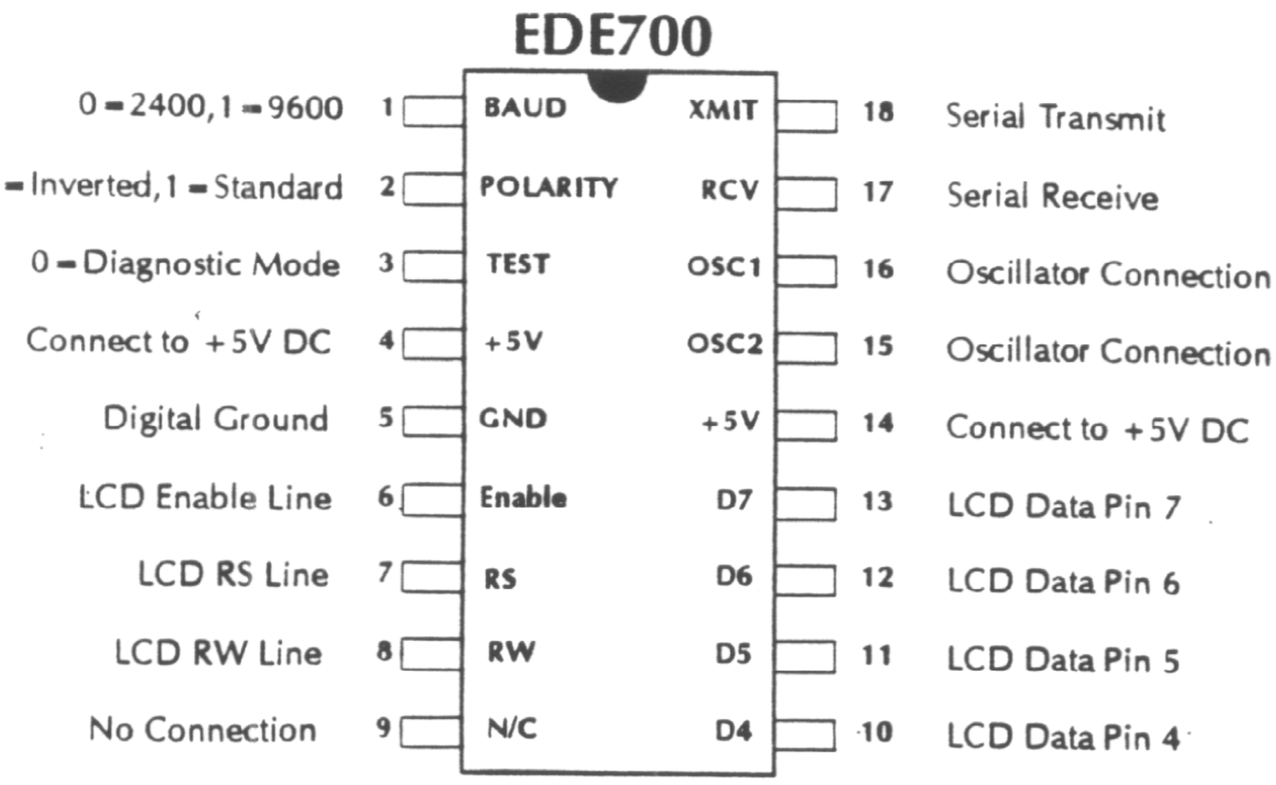
141559

EDE700 Serial LCD Interface IC



The EDE700 Serial LCD Interface IC is designed to conveniently control text-based LCD modules via a simple two or three wire link. This IC allows both engineer and hobbyist alike to easily incorporate a LCD module into their design using a minimal amount of I/O lines. The EDE700 allows full LCD control, including the creation of custom characters. Also, in addition to the usual sideways-scrolling feature standard on most LCD modules, the EDE700 is also capable of shifting text from the bottom row of a 2-row LCD to the top row using only one command. 2400 or 9600 Baud rates and selectable serial data polarity allow the EDE700 to communicate easily with most serial devices. Connection to most microcontrollers, stamps, or microprocessors can be made using a single data wire, without any type of voltage level conversion. Connection to a PC can be made using only a 33KΩ resistor.

Some of the special features of the EDE700 are as follows:

- ***RS232 and TTL-level compatible***
- ***Single-chip design allows for easy system integration***
- ***Compatible with most 1x8 to 2x40 character HD44780 controlled LCDs***
- ***Useable with LCDs having either 1x14 or 2x7 connector configuration***
- ***Allows direct control of all LCD functions***
- ***Adds new "Row Shift-up" capability***
- ***Selectable data polarity eliminates necessity of RS-232 voltage converter***
- ***Data Read Mode allows screen data to be read back to host system***
- ***2400/9600 Baud data rates***
- ***Special Diagnostic Mode tests screen hookup and data transmission***
- ***Available in 18 pin DIP or SOIC packages***

Copyright © 1997 E-Lab Digital Engineering, Inc. All Rights Reserved.

PIN DEFINITIONS

LCD Connection Pins

LCD Enable Line (Pin 6): ..... LCD Enable line;
pin 6 on LCD screen module

LCD RS Line (Pin 7): ..... LCD Register Select line;
pin 4 on LCD screen module

LCD RW Line (Pin 8): ..... LCD Read/Write line;
pin 5 on LCD screen module

D4 (Pin 10): ..... LCD Data 4 line;
pin 11 on LCD screen module

D5 (Pin 11): ..... LCD Data 5 line;
pin 12 on LCD screen module

D6 (Pin 12): ..... LCD Data 6 line;
pin 13 on LCD screen module

D7 (Pin 13): ..... LCD Data 7 line;
pin 14 on LCD screen module

EDE700 Control & Data Pins

BAUD (Pin 1): ..... Baud Rate Selection;
0 = 2400, 1 = 9600 Baud

Polarity (Pin 2): ..... Serial Data Polarity Control;
0 = Inverted, 1 = Standard

Test (Pin 3): ..... Diagnostic Mode;
0 = Test Mode, 1 = Normal Operation

RCV (Pin 17): ..... Serial Data Input

XMIT (Pin 18): ..... Serial Data Output (Normally unused)

Clock/ Power Pins

OSC1,OSC2 (Pin 16, Pin 15): ..... 20 MHz Resonator Connection

+5V Power (Pin 14, Pin 4): ..... Connect to +5V DC

Ground (Pin 5): ..... Connect to 0 VDC (GND)

PC CONNECTION

The EDE700 can be used with most HD44780-based text LCD's. The two most common types of connectors on these modules are the 1x14 and 2x7 pin arrangements (there may be two additional pins if the LCD is the backlit type). The pin numbers should be identified on the LCD module. Refer to Figure One for connection of the LCD module to the EDE700. Notice that not all of the LCD module's pins need to be connected, as the EDE700 uses a 4-bit transfer mode.

Figure One also illustrates the hookup used for connection to a PC. A RS-232 voltage-level converter IC such as the MAX232 is not required; internal clamping diodes on the EDE700 convert the incoming RS-232 level voltages into TTL-level signals. Also, notice that the POLARITY input on the EDE700 is tied low; this causes the EDE700 to interpret the inverted RS-232 signals correctly (a voltage-converter IC such as the MAX232, inside the PC, inverts the signal during transmission).

*** IMPORTANT NOTE:** *The 33KΩ resistor must be used when connecting to a PC in this fashion; omitting it will cause excessive current to flow, possibly damaging both the PC serial port and the EDE700 IC.*

Copyright © 1997 E-Lab Digital Engineering, Inc. All Rights Reserved.

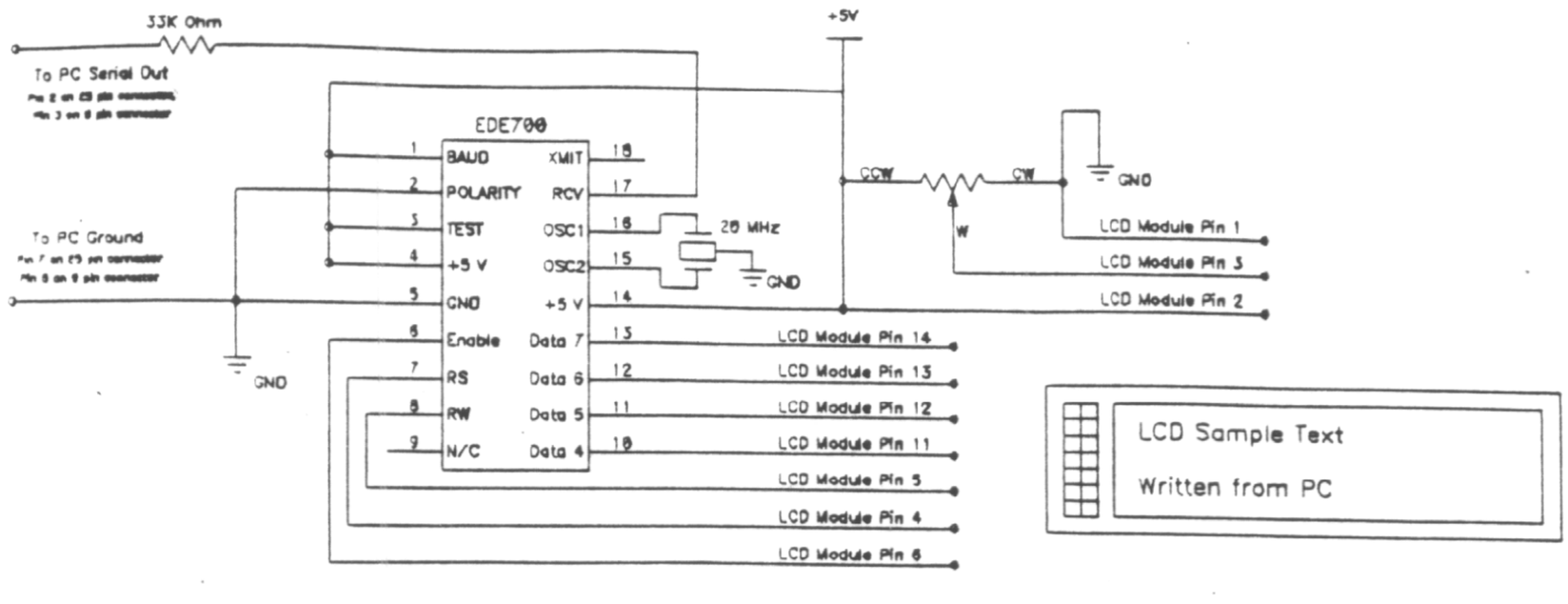


Figure One: Connection of EDE700 to a PC using only a 33KΩ Resistor

A 20MHz resonator is used to clock the EDE700. Alternately, you may choose to use a 20MHz crystal or a 20MHz TTL oscillator. If you choose the latter, connect the oscillator output to only OSC1 (Pin 16). The LCD contrast is set by a 10-20KΩ potentiometer. If the LCD screen will be used in a relatively temperature-constant environment with a fixed viewing angle, you may choose to replace this potentiometer with two suitably-valued resistors connected in series. The contrast pin should be connected between the two resistors (use a potentiometer to first determine the appropriate resistance values for your particular LCD).

The above schematic can be easily tested using the following QBASIC code (both the compiled and the text version are available on E-Lab's web site):

```
REM Open channel to COM2 at 9600 Baud
OPEN "com2:9600,n,8,1,cd0,cs0,ds0,op0,rs" FOR OUTPUT AS #1

REM Clear Display
OUT &H2F8, &HFE
GOSUB 999
OUT &H2F8, &H1
GOSUB 999

REM Write first row of text to LCD screen
PRINT #1, "EDE700 Test Screen";
GOSUB 999

REM Jump to second row on 2 line LCD
OUT &H2F8, &HFE
GOSUB 999
OUT &H2F8, &HC0
GOSUB 999

REM Write second row of text to LCD screen
PRINT #1, "Time is: "; TIME$;

END

REM Adjust delay value according to PC speed.
REM 50000 is for Pentium; XT would use much lower figure
999 FOR delay = 1 TO 50000: NEXT delay
RETURN
```

Notice that the above code uses the COM2 port; to use the COM 1 port modify the "OPEN" line to read "COM1" instead and also change the address &H2F8 to &H3F8 on the "OUT" lines.

Copyright © 1997 E-Lab Digital Engineering, Inc. All Rights Reserved.

DIAGNOSTIC MODE

When initially connecting the EDE700, you may wish to verify that you have the EDE700 to LCD screen interface correctly wired prior to connecting the controlling device (PC, microcontroller, etc.). A special diagnostic mode has been built into the EDE700 for this purpose.

Connect the LCD screen as illustrated in Figure One. The PC connection may be left undone for now. Before turning on power to the circuit, connect the EDE700's TEST Pin (Pin 3) to Ground instead of +5V. This will instruct the EDE700 to power up in Diagnostic Mode. When you apply power, the message "EDE700 Version 1.00,N-8-1 (C) E-Lab, Inc" should appear on your display screen (assuming your screen is wide enough to view the entire message). If this message appears on your screen, the LCD module is connected correctly to the EDE700. If it does not appear, turn the power off and double-check your wiring, including LCD module power and ground. (*Don't forget to try adjusting your contrast potentiometer - the display contrast might be the only problem!*) If you can't find the problem, consult the manufacturer's datasheet for your particular LCD module. Although unlikely, it may have a custom pinout. The EDE700 will work with most any LCD module that utilizes the HD44780 controller.

In addition to displaying the above message on the screen, starting the EDE700 in Diagnostic Mode also causes a stream of "A"s to be sent out the Serial Transmit pin (Pin 18). This allows the connection of the serial data stream from the EDE700 back to the host to be verified. This connection is not necessary for ordinary operation and should only be connected if LCD screen data will need to be read back from the LCD display module. Ordinarily, just two pins from the EDE700 will be connected to the host system - Serial Receive (Pin 17) and Ground (Pin 5).

CONNECTION TO OTHER SYSTEMS

The EDE700 can be connected to a variety of host controllers other than a PC, including the BASIC Stamp™ series, numerous microcontrollers, or a microprocessor system. The type of connection used is determined by the communications level voltage of the host system. For instance, the PC serial port transmits using voltages ranging from -15 to +15 VDC. The schematic in Figure One relies on the EDE700's internal input-clamping diodes to reduce the signal's voltage to TTL-levels. The 33KΩ resistor is necessary to limit the current through these diodes. Because the voltage driver inside the PC contains a level inverter, we use the EDE700 in "Inverted Mode" (Pin 2 Low). This causes the EDE700 to transmit and receive using a polarity-inverted serial data stream.

If you do choose to use a voltage-converter IC such as the MAX232 (for instance, if the host is separated from the EDE700 by a considerable distance), do not also use the 33KΩ resistor. Since the MAX232 Level Shifter IC contains an inverter, the EDE700 should, when used in this configuration, be operated in "Standard Mode" (Pin 2 High). In addition, if the EDE700 is to write data back to the PC (used only when LCD character data needs to be read from the LCD screen), you must use the MAX232 or similar voltage level converter to generate the RS-232 signal voltages that the PC needs to operate properly. Again, in this case, operate the EDE700 in "Standard Mode". Data Readback Mode used with stamps or microcontrollers, however, does not require the use of a level-converter IC, as can be seen in Figure Three.

Connection to a BASIC Stamp™ or microcontroller requires no voltage level shifter or current-limiting resistor, simply connect the microcontroller's serial output to the EDE700's Serial Data Input (Pin 17). Depending upon the length of the data line, you may need to pull this line high using a 4.7KΩ resistor, as shown in Figure Two. If your design needs to read data back to the host microcontroller from the LCD, connect the EDE700's Serial Data Output (Pin 18) to the microcontroller's serial data input. Typically, an arrangement such as this would use the EDE700 in "Standard Mode" (serial data not inverted). Figure Two illustrates the connection of a BASIC

Copyright © 1997 E-Lab Digital Engineering, Inc. All Rights Reserved.

Stamp™ I to the EDE700. Notice that only one I/O pin is required to display text on the LCD screen. For this example, the EDE700 is set to 2400 Baud (Pin 1 Low), Standard Polarity (Pin 2 High).

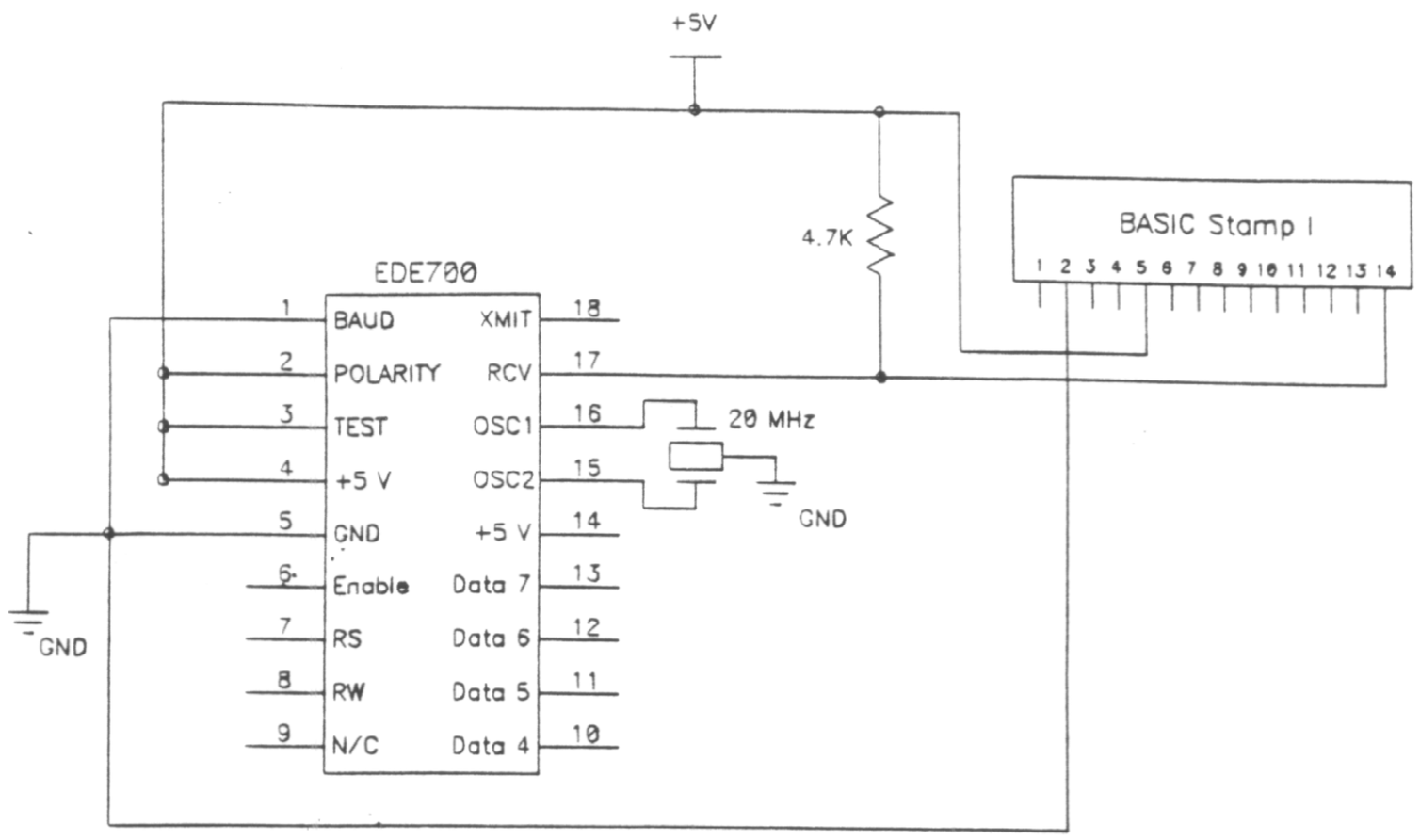


Figure Two: Connection of EDE700 to BASIC Stamp™ I

The following software, written for the BASIC Stamp™ I, will output text on both lines of a two-line LCD. Notice that a "Pause 10" instruction is added after each write to the LCD internal control to allow time for the LCD to execute the instruction.

```
REM Delay while LCD powers up...
PAUSE 500

REM Clear display and home cursor
SEROUT 7,T2400,($FE)
SEROUT 7,T2400,($01)
PAUSE 10

SEROUT 7,T2400,("Hello World...")

REM Move to start of second row
SEROUT 7,T2400,($FE)
SEROUT 7,T2400,($C0)
PAUSE 10

SEROUT 7,T2400,("This text should be on the second row!")
```

The following section outlines the use and function of the EDE700 LCD controller. While this datasheet explains many of the specific LCD module features, your particular module may be slightly different. When in doubt, always refer to the datasheet for your particular LCD module. In addition, we have found that some one-line LCD modules do not have the entire row mapped contiguously in memory; you may have to move the cursor position to the start of the LCD memory's "second line" in order to write to the last half of the characters on a one-line display.

Copyright © 1997 E-Lab Digital Engineering, Inc. All Rights Reserved.

USING THE EDE700 SERIAL LCD CONTROLLER

The EDE700 can be written to in much the same way as a terminal display device. It accepts standard ACSII characters and causes them to be written to a display screen. In addition to this, the EDE700 will interpret several ASCII control commands and perform specific functions as directed by them. This section details the use of the various commands available on the EDE700.

After powerup, your host controller (the device writing to the EDE700) should wait approximately 50 mS for the LCD to initialize. Following this, the display is ready to accept data for display. Before the first character is written, the EDE700 will home the cursor (the cursor will be invisible). Therefore, the first character written to the EDE700 will be displayed in the upper-left hand corner of a two row display, or in the leftmost position of a one-row LCD. Subsequent character writes to the EDE700 will cause the cursor location to increment by one for each character written.

Most ASCII characters can be displayed on the LCD module using the EDE700 (see your LCD module datasheet for a complete listing). However, the EDE700 will intercept certain ASCII control characters and not display them to the screen, instead performing a specific command on the LCD module. The following table shows the ASCII codes that will be intercepted by the EDE700:

Value	ASCII Command	Function Performed
$0D	Carriage Return	If previous character was line feed, print nothing. Otherwise, shift row 2 to row 1 and put cursor at start of row 2.
$0A	Line Feed	If previous character was a carriage return, print nothing. Otherwise, shift row 2 to row 1 and put cursor at start of row 2.
$0C	Form Feed	Clear screen, home cursor.
$08	Backspace	Perform destructive backspace (move cursor back one position, and erase character in that position).
$FE	EDE700 Control	Read next command and perform desired function.

Table One: ASCII Control Characters

As can be see from Table One, either a Carriage Return (CR) or a Line Feed (LF) sent to the EDE700 causes it to copy the text on the second row of the LCD module to the top row, clear the second row, and place the cursor at the start of the second row. Because many systems send both the CR/LF combination at the end of a row of text, the EDE700 will only act upon one of these two commands if they are sent one immediately following the other. Your host system may need to delay a brief period while this action is performed.

Sending the Form Feed character to the EDE700 causes it to clear the screen and home the cursor.

The Backspace command causes the cursor to be moved back by one space (on a given row), deleting the character that was in that position (commonly called a "destructive backspace").

Notice that the last command, $FE, has special meaning to the EDE700. It is used to issue direct commands to the LCD module. The following table shows the direct commands that can be sent after sending the value $FE (1111 1110 binary, 254 decimal) to the EDE700:

Copyright © 1997 E-Lab Digital Engineering, Inc. All Rights Reserved.

Binary Value Sent	Action Taken
0 0 0 0 0 0 0 1	Clear display and return cursor to home position.
0 0 0 0 0 0 1 X	Home Cursor.
0 0 0 0 0 1 I/D S	Sets cursor move direction & shifting. I/D=1 for increment,S=1 for shift.
0 0 0 0 1 D C B	Sets display on (D), cursor on (C), blink on (B). 1=on, 0=off.
0 0 0 1 S/C R/L X X	Moves cursor and shifts display without affecting DDRAM. S/L=1: display shift; 0=cursor movement. R/L=1, shift right; 0=left.
0 0 1 DL N F X X	Sets data length (DL), number of lines (N), and character font (F). DL=1:8 bit; 0=4 bit. N=1:2 lines;0=1 line. F=1:5x10;0=5x7 dots.
0 1 X X X X X X	Set CGRAM Address. Character data is sent after receiving this setting.
1 X X X X X X X	Set DDRAM Address. Data display address is set using this command. (Line one starts at $00, line two starts at $40 (Therefore the command $C0 would move the cursor to the start of the second row).
1 1 1 1 1 1 1 0	Special EDE700 command; used to read data at a specific LCD address. See text for details.

Table Two: $FE Direct Control Codes

As an illustration of these direct control codes, the following code, written in QBASIC, will print two lines of text on the LCD and cause the cursor to blink. It then scrolls the entire display to the left 40 positions.

```
DECLARE SUB pause ()
OPEN "com2:9600,n,8,1,cd0,cs0,ds0,op0,rs" FOR OUTPUT AS #1

REM Wait while LCD powers up
GOSUB 999

REM Clear Display
OUT &H2F8, &HFE: GOSUB 999
OUT &H2F8, &H1: GOSUB 999

PRINT #1, "Line One Text."; : GOSUB 999

REM Move to first location on row two
OUT &H2F8, &HFE: GOSUB 999
OUT &H2F8, &HC0: GOSUB 999

PRINT #1, "Line Two Text."; : GOSUB 999

REM Shift Entire Display Left
FOR shift = 1 TO 40
OUT &H2F8, &HFE: GOSUB 999
OUT &H2F8, &H18: GOSUB 999
NEXT shift

END

REM Adjust delay value according to PC speed.
REM 50000 is for Pentium; XT would use much bwer figure
999 FOR delay = 1 TO 50000: NEXT delay
RETURN
```

The last command listed in Table Two allows data being displayed on the LCD to be read back into the controlling device. Although not commonly used in most designs, this feature can be used to read variables or other data back into the host. The 8-bit value is returned to the host via the EDE700's XMIT pin (Pin 18), as mentioned in a previous section. Upon issuing this command, the EDE700 waits for a row value and a column value before transmitting the character at that position back to the host. The following schematic illustrates a PIC16C84 being used to write text to the LCD module's first row, read it back, and write it again to the second row.

Copyright © 1997 E-Lab Digital Engineering, Inc. All Rights Reserved.

Note: The LCD cursor position remains at the last location data was read from. Therefore, subsequent data written to the LCD may not be placed at the position expected unless the LCD cursor position is reset following a data read, as illustrated in the code below.

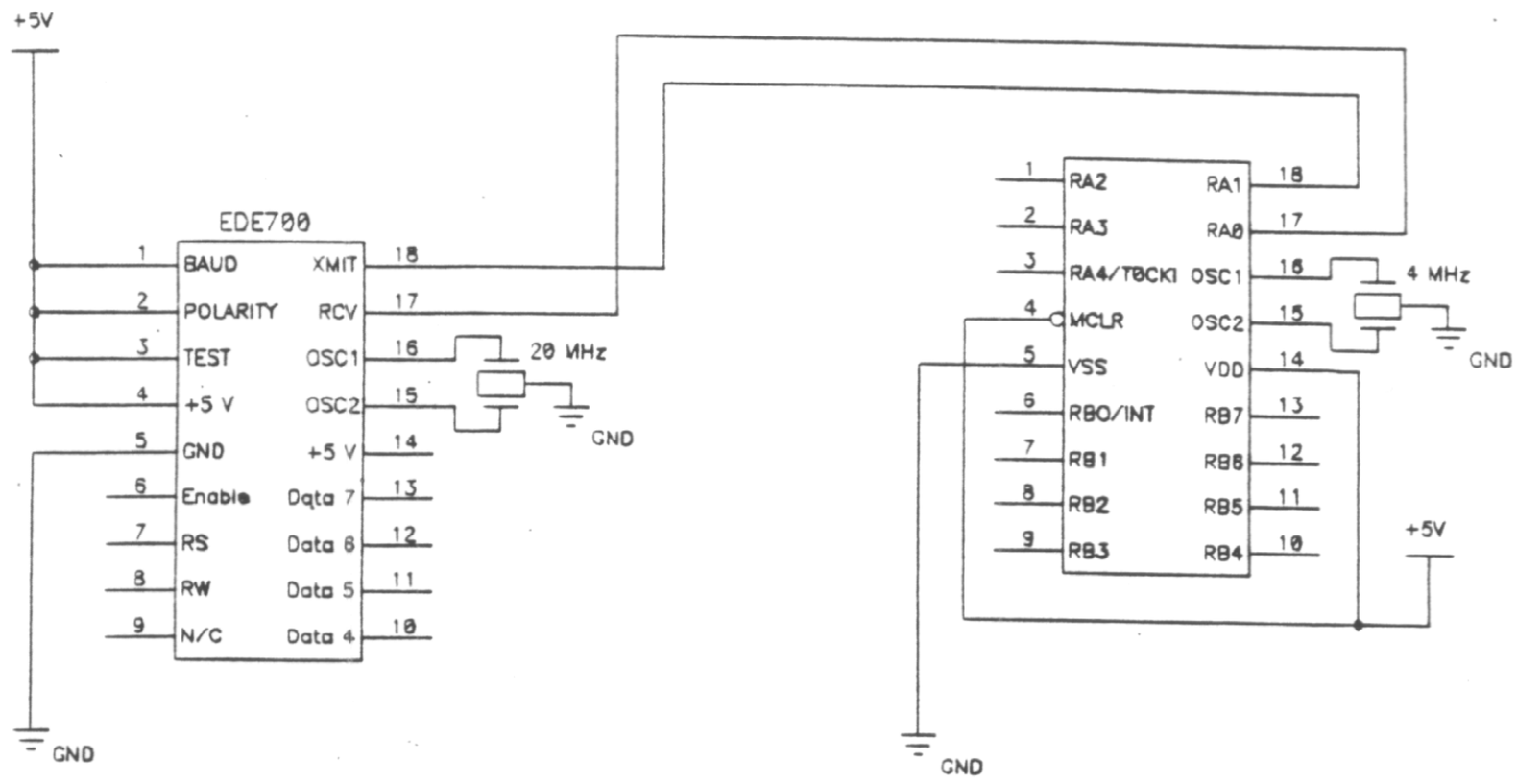


Figure Three: Interface to PIC16C84, Using Write and Read Connections

This code, written in C (PCM Compiler from Custom Computer Services, Inc.), illustrates the use of the data readback command in conjunction with the above schematic. As mentioned earlier, this simple example merely echoes the top row's text to the second row of a two-line LCD. A practical use for this feature would be the reading of variable data back from the LCD in limited-memory systems.

```
#include<16C84.H>
#use delay(clock=4000000)
#use rs232(baud=9600,rcv=pin_a1,xmit=pin_a0)

main() {

char value;
int column;

delay_ms(50);                          /* Wait while LCD powers up */

printf("Line Number One...");          /* Write text to first row */

for (column=1;column<=40;column++)
{
putc(0xFE);                                  /* Enter Direct Control Mode */
putc(0xFE);                                  /* Select 'Data Readback' Function */
putc(0x01);                            /* Select Row 1 */
putc(column);                          /* Select Column */
value = getc();                        /* Read Character */
putc(0xFE);                                  /* Enter Direct Control Mode */
putc(0xC0 + column -1);                /* Move cursor to appropriate position on second row */
putc(value);                           /* Write Character on Second Row */
}

}
```

Copyright © 1997 E-Lab Digital Engineering, Inc. All Rights Reserved.

CREATING CUSTOM SCREEN CHARACTERS

You may occasionally have need to display a character not included in the standard ASCII character set. Or, as is the case in this example, you may need to modify a certain character to make it more suitable for your application. In this instance, we choose to make a "\" because the backslash character on most LCD modules is an unusable, odd-looking character. A BASIC Stamp™ is used as the controlling host, as was illustrated in Figure Two.

Because the Hex Codes $00 through $07 are seldom used, you should use these locations to store your custom characters. Each character is displayed as an 8 row by 5 column grid. Grid data is written to the CGRAM memory area of the LCD Module (see Table Two for details). Because each character contains eight bytes of data, and because the CGRAM area starts at $40 (0100 0000 binary, again see Table Two), the CGRAM address for ASCII character $00 is $40, ASCII character $01 is $48, ASCII character $02 is $50, etc.

The 8x5 grid is arranged in CGRAM memory as shown below:

```
---X XXXX   ($1F, or 0001 1111 binary, would turn on all the pixels in this row)
---X XXXX
---X XXXX
---X XXXX
---X XXXX
---X XXXX
---X XXXX
---X XXXX
```

The "X" locations are turned on or off depending upon whether a 0 or a 1 is written to that particular location in memory. For instance, to turn all the pixels on in the top row of a character, we would write the value $1F (0001 1111 binary) to the CGRAM. However, to prevent the EDE700 from intercepting certain ASCII values as commands and interpreting them as shown in Table One, we will perform a logical "or" operation on all pixel data that is sent to the CGRAM. The value that should be "or"ed with the pixel data is $20. Therefore, instead of sending $1F, we should send $3F (0011 1111 binary). All pixel data in the following code has been logically "or"ed with $20.

The following BASIC Stamp™ I code causes the ASCII character $03 to be displayed as a "\" character on the LCD module:

```
REM Wait while LCD powers up
PAUSE 200

REM Set CGRAM address to character $3: $40 + $8 + $8 + $8 = $58
SEROUT 7,T2400,($FE)
SEROUT 7,T2400,($58)

REM Enter data for grid. Example: '---1 0000' on first row attained
REM  by sending $10 (0001 0000 binary)
REM NOTE: To prevent the EDE700 from intercepting certain ASCII value
REM  and interpreting them as commands, all grid values are "or"ed with $20
REM  For example, $10 'or'ed (+) with $20 = $30
SEROUT 7,T2400,($30)
SEROUT 7,T2400,($30)
SEROUT 7,T2400,($28)
SEROUT 7,T2400,($24)
SEROUT 7,T2400,($22)
SEROUT 7,T2400,($21)
SEROUT 7,T2400,($21)
SEROUT 7,T2400,($20)

REM Move cursor back to start of DDRAM area
SEROUT 7,T2400,($FE)
SEROUT 7,T2400,($80)

REM Write newly-defined character
SEROUT 7,T2400,($03)
```

Copyright © 1997 E-Lab Digital Engineering, Inc. All Rights Reserved.

Although we strive to maintain a high degree of technical accuracy, this document may contain typographical errors or other inaccuracies. We reserve the right to correct any inaccuracy in future editions of this document without notice. This document and examples contained herein are provided 'As Is' without warranty of any kind expressed or implied. This document is to be used for informational purposes only and may not be duplicated for commercial purposes other than by distributors authorized by E-Lab.

ABSOLUTE MAXIMUM RATINGS

Oscillator frequency ..... 20 MHz
Supply Voltage 7.5V
Ambient temperature under bias ..... -40°C to +125°C
Max. current sinked by output pin 25mA
Max. current sourced by output pin 25mA

STANDARD OPERATING CONDITIONS

Supply voltage ..... 3.0V to 5.5V
Operating temperature ..... 0°C to +70°C

The EDE700 IC is implemented as firmware on a PIC16C621 microcontroller, manufactured by Microchip Technology, Inc. For a more comprehensive technical summary of this device, please refer to the PIC16C621 datasheet (accessable from the E-Lab web site).

IMPORTANT NOTICE

E-LAB Digital Engineering, Inc. (E-LAB), reserves the right to change products or specifications without notice. Customers are advised to obtain the latest versions of product specifications, which should be considered when evaluating a product's appropriateness for a particular use.

THIS PRODUCT IS WARRANTED TO COMPLY WITH E-LAB'S SPECIFICATION SHEET AT THE TIME OF DELIVERY. BY USING THIS PRODUCT, CUSTOMER AGREES THAT IN NO EVENT SHALL E-LAB BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES AS A RESULT OF THE PERFORMANCE, OR FAILURE TO PERFORM, OF THIS PRODUCT.

E-LAB MAKES NO OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

E-LAB's LIABILITY IS FOR A PERIOD NO GREATER THAN 90 DAYS FROM DATE OF SHIPMENT BY E-LAB AND IS LIMITED TO REPLACEMENT OF DEFECTIVE PRODUCT. This warranty covers only defects arising under normal use and not malfunctions resulting from misuse, abuse, modification, or repairs by anyone other than E-LAB.

E-LAB'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF E-LAB. Life support devices or systems are those which are intended to support or sustain life and whose failure to perform can be reasonably expected to result in a significant injury or death to the user. Critical components are those whose failure to perform can be reasonably expected to cause failure of a life support device or system or affect its safety or effectiveness.

COPYRIGHT NOTICE

This product may not be duplicated. E-LAB Digital Engineering, Inc. holds all copyrights on firmware, with all rights reserved. Unauthorized duplication of this device may be subject to penalty under state and/ or federal law.

EDE700 and the E-LAB logo are trademarks of E-LAB Digital Engineering, Inc. All other trademarks and registered trademarks are property of their respective owners.

CONTACTING US

We are continually updating our product line. Please contact us for our latest product information.